Проект по предметот

Агентно базирани системи

Тема:

Deep Q-Learning за играта Snake

Дарко Диванисов 201023

Февруари 2026

Содржина

1. Апстракт

Овој труд го претставува развојот и евалуацијата на агент за учење со потикнување (Reinforcement Learning) кој учи да ја игра класичната игра Snake користејќи го Deep Q-Learning (DQN) алгоритмот. Агентот користи невронска мрежа со повеќе слоеви за да ги процени Q-вредностите за секоја можна акција во дадена состојба на играта.

Имплементиран е directional тип на репрезентација на состојбата кој ги вклучува информациите за опасностите во близина на главата на змијата, насоката на движење, локацијата на храната и должината на телото.

Експериментите се спроведени со различни конфигурации на хиперпараметри, вклучувајќи големина на batch (128, 256, 500) и големина на меморија (2048, 4096, 8192, 50000).

2.Вовед

Играта Snake е класичен проблем во областа на вештачката интелигенција кој претставува предизвик за агенти за засилено учење. Играта бара од агентот да научи стратегии за:

- Избегнување на судир со ѕидовите и сопственото тело
- Ефикасно наоѓање на храната
- Планирање на пат за да се избегне затворање
- Управување со растечкото тело на змијата

Проблемот станува потешок како што змијата расте, бидејќи слободниот простор се намалува, а агентот мора да планира подолгорочно за да избегне патеки кои водат кон смрт.

3.Сродни истражувања

Deep Q-Networks (DQN) алгоритмот за прв пат е воведен од страна на Mnih (2015) од DeepMind, кој покажа дека комбинацијата на Q-learning со длабоки невронски мрежи може да постигне човечки или супер-човечки перформанси во Atari игри. Нивната работа ги воведе клучните техники како experience replay и target networks.

За специфично играта Snake, постојат повеќе имплементации со различни пристапи:
Patel et al. (2020) имплементираа DQN со CNN архитектура која го обработува целиот екран на играта. Нивниот пристап постигна добри резултати но бараше значително повеќе компонентачки ресурси.
Liu & Zou (2019) користеа поедноставна репрезентација со 11-те бинарни вредности слична на мојата directional репрезентација, но без дополнителните информации за должината и слободниот простор.
Во споредба со сродните истражувања, мојот пристап воведува неколку новини:

1. Проширена directional репрезентација со 18 карактеристики

2. Инкорпорирање на информација за должината на телото
3. Анализа на слободниот простор во сите насоки
4. Двофазно тренирање со различни epsilon decay стратегии

4. Опис на агентот и методите

4.1 Архитектура на невронската мрежа

За directional репрезентацијата е користена невронска мрежа со следната архитектура:

Слој	Неврони	Активација
Влезен	18	-
Скриен 1	256	ReLU
Скриен 2	128	ReLU
Скриен 3	64	ReLU
Излезен	3	-

Мрежата има 18 влезни неврони согласно со големината на directional состојбата од самата игра која е имплементирана со библиотеката pygame и излез со 3 неврони кои ги претставуваат акциите: оди право, сврти на десно и сврти на лево.

За истражување се користи epsilon-greedy стратегија каде epsilon се намалува експоненцијално од 1.0 до 0.03 со decey фактор од 0.9985. Во втората фаза на дотренирање epsilon се намалува од 0.03 до речиси 0 со побрз decey од 0.99.

4.2 Репрезентација на состојбите

Directional репрезентација на состојбата вклучува низа со должина од 18 броеви

	Карактеристика	Опис
1-3	Опасност напред/лево/десно	Бинарно (0 или 1)
4-7	Насока на движење	Бинарно(one-hot encoded)
8-9	Тело лево/десно	Бинарно(0 или 1)
10	Должина на телото	Нормализирано(со максималната потенцијална должина возможна)
11-14	Слободен простор	Нормализирано(0-1)
15-18	Локација на храна лево/десно/горе/доле	Бинарно(0 или 1)

4.3 Систем за награди

Системот за награда е дефиниран со следните вредности:

- Јадење храна (+10)

- Судир/смрт (-10)
- Нормален чекор (0)

Дополнително е имплементиран механизам за рано сопирање во тренинг фаза ако поминале 100 чекори од кога последно змијата изела храна но само ако змијата е пократка од 10 во спротивно го нема овој механизам, причина да се спречи циркуларно лутање на змијата во раните фази на играта кога змијата е пократка, а пак во тест фазата е имплементирано сопирање ако се поминале 1000 чекори без змијата да изеде храна. За разлика од тренинг фазата каде моделот е казнет со (-10) за вакво движење тест фазата нема казна бидејки негативно влијае врз ресултатот.

5. Експериментални резултати од тренинг фазата

5.1 Параметри на тренирање

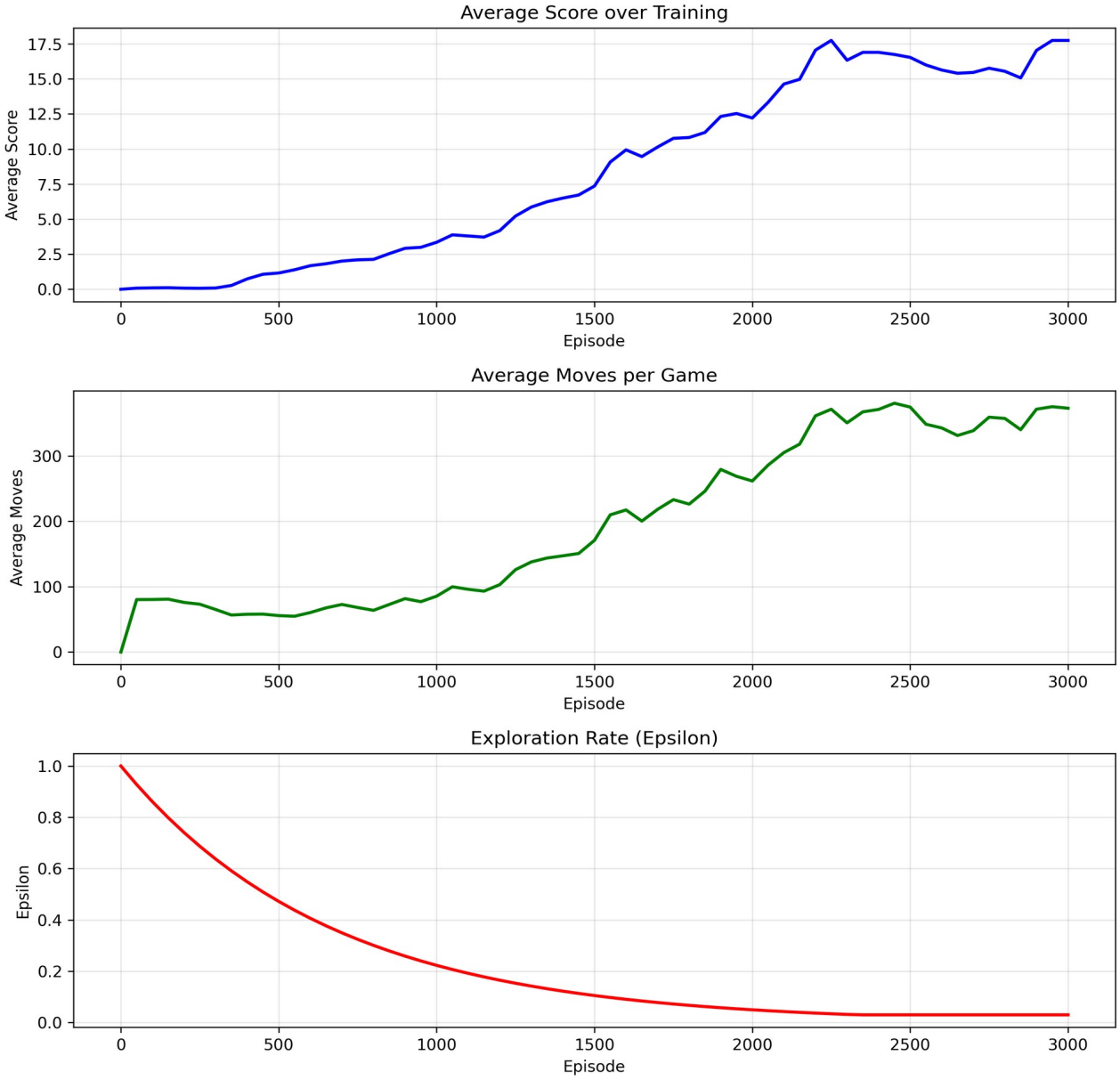
Спроведени се експерименти со различни конфигурации на хиперпараметри. Следната табела ги прикажува конфигурациите.

Име на агентот	Batch	Memory	LR	Episodes	Epsilon range
Агент 1	128	2048	0. 001	3000	Од 1 до 0.03
Агент 2	256	4096	0. 001	3000	Од 1 до 0.03
Агент 3	256	8192	0. 001	3000	Од 1 до 0.03
Агент 4	512	8192	0. 001	3000	Од 1 до 0.03
Агент 1-v2	128	2048	0. 001	1000	Од 0.03 до 0
Агент 2-v2	256	4096	0. 001	1000	Од 0.03 до 0
Агент 3-v2	256	8192	0. 001	1000	Од 0.03 до 0
Агент 4-v2	512	8192	0. 001	1000	Од 0.03 до 0

Спроведено е тренирање на секоја конфигурација по двапати првиот пат е спроведена во 3000 епизоди со опаѓање на епсилон од 1 до 0.03 а второто тренирање(дотренирање) е спроведено на истиот агент но во 1000 епизоди со епсилон од 0.03 па се до 0. Ова е направено на овој начин со цел да се избегне огромно тренирање наеднаш и исто така побитно да се види дали агентот ќе се подобри доколку тренира неколку стотици епизоди со приближно вредност 0 за епсилон и затоа агентот од рунда 1 и рунда 2 се зачувани како посебени агент.

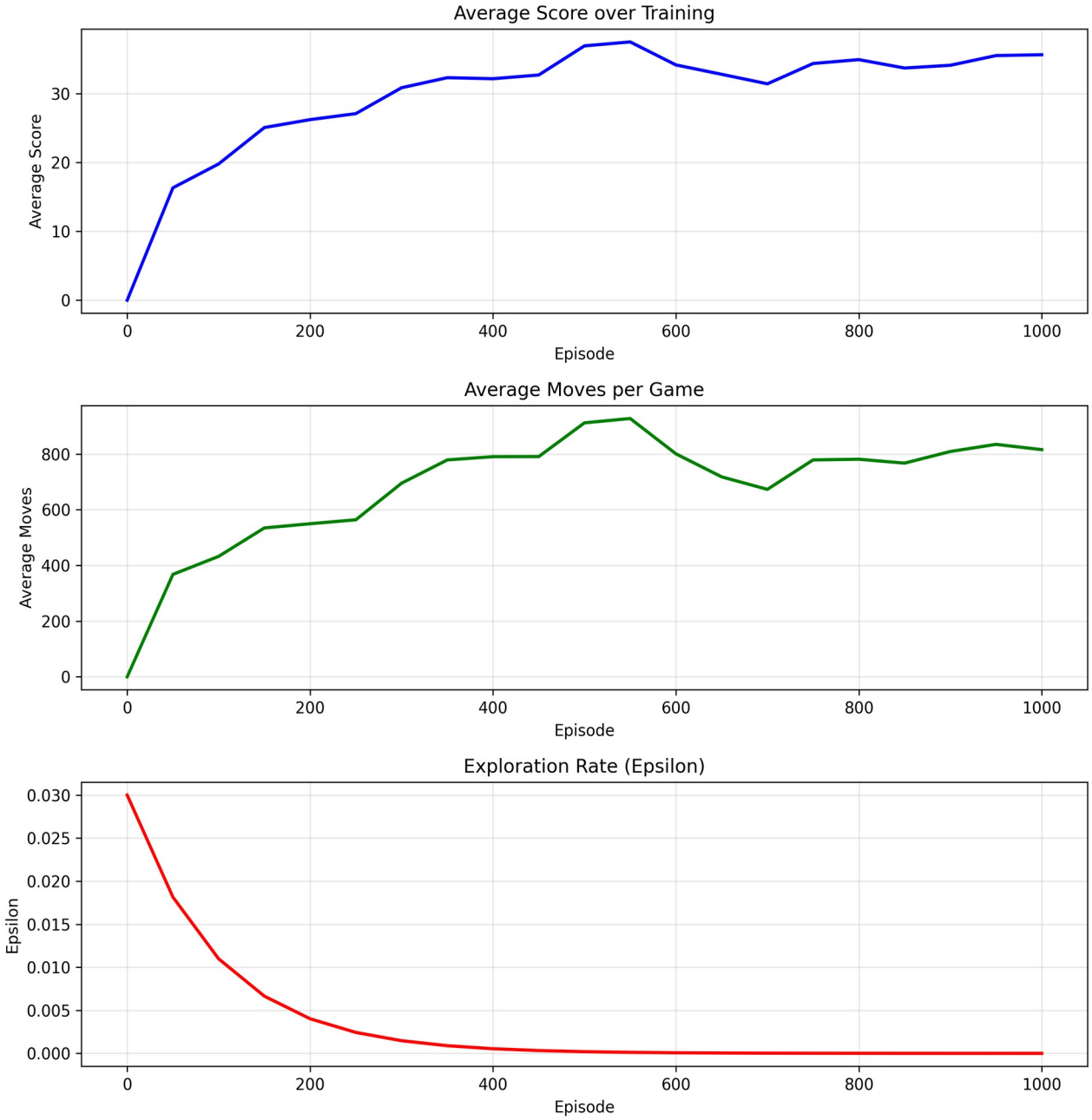
5.2 Графици за тренинг фазата

Следуваат графици од тренинг фазата на секој агент. За секој агент има три линиски графици кој прикажуваат просечен резултат, просечен број чекори и епсилон на секој 50 чекори.



сл. бр. 1 Агент 1

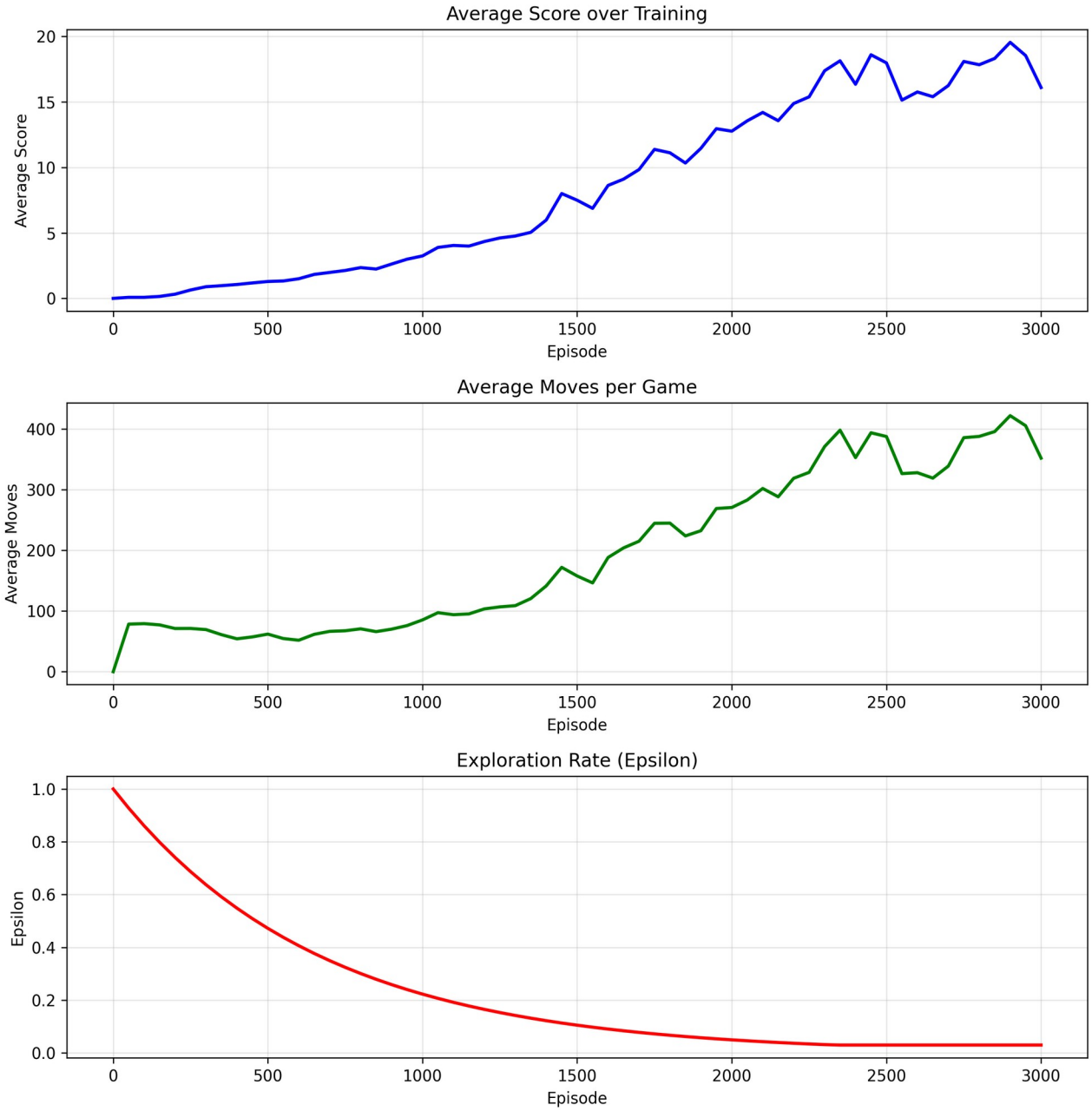
Агентот 1 со хиперпараметри batch 128 и memory 2048, овој агент е најпрвиот агент кој е трениран, резултатите при тренирање се следните околу епизода 2200 достигнува 17.5 просек



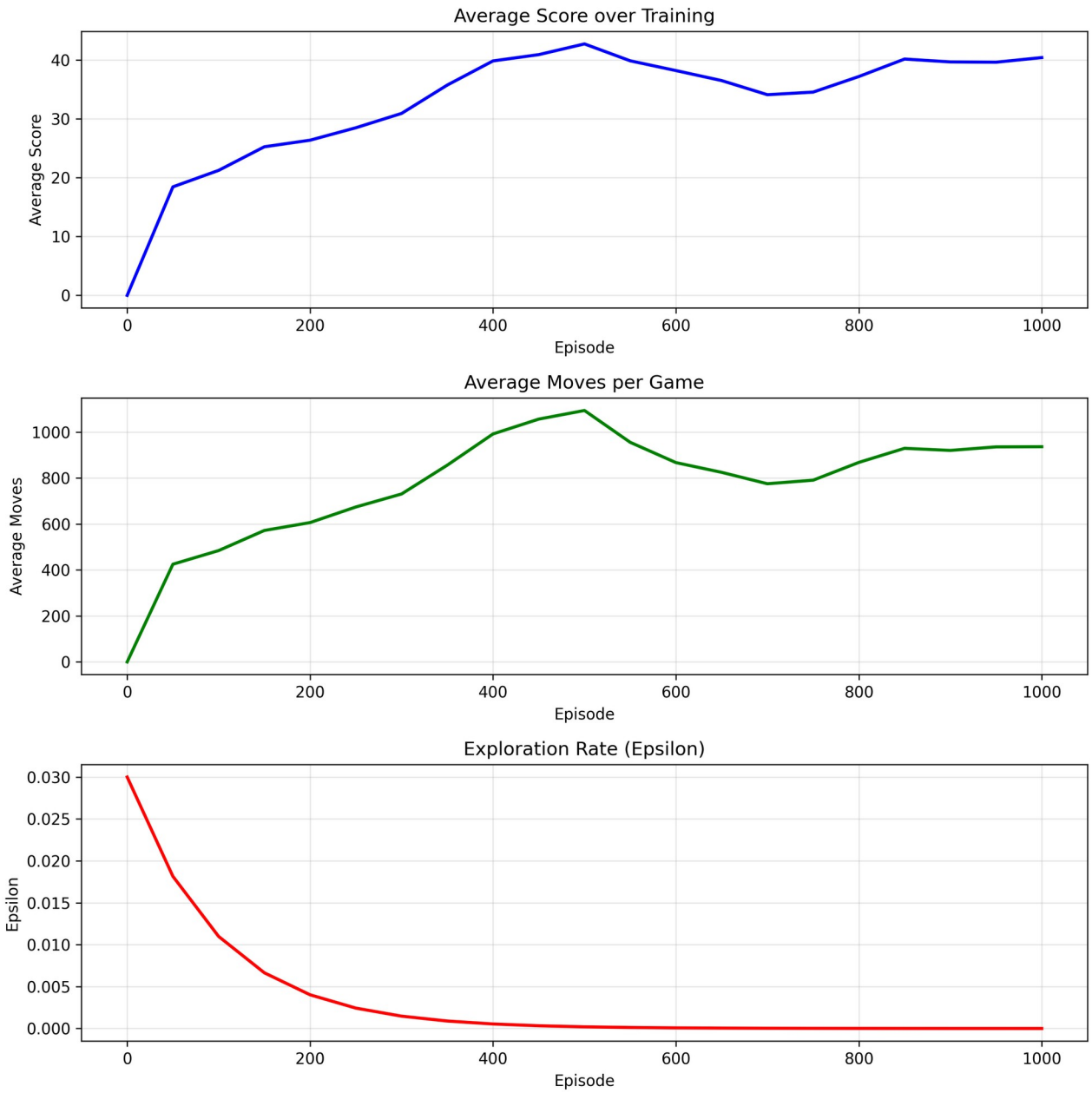
резултат, соброј на чекори при игра околу 350. Е поставена почетна линија која се надевам ке ја надминат другите агенти.

сл. бр.2 Агент 1-v2

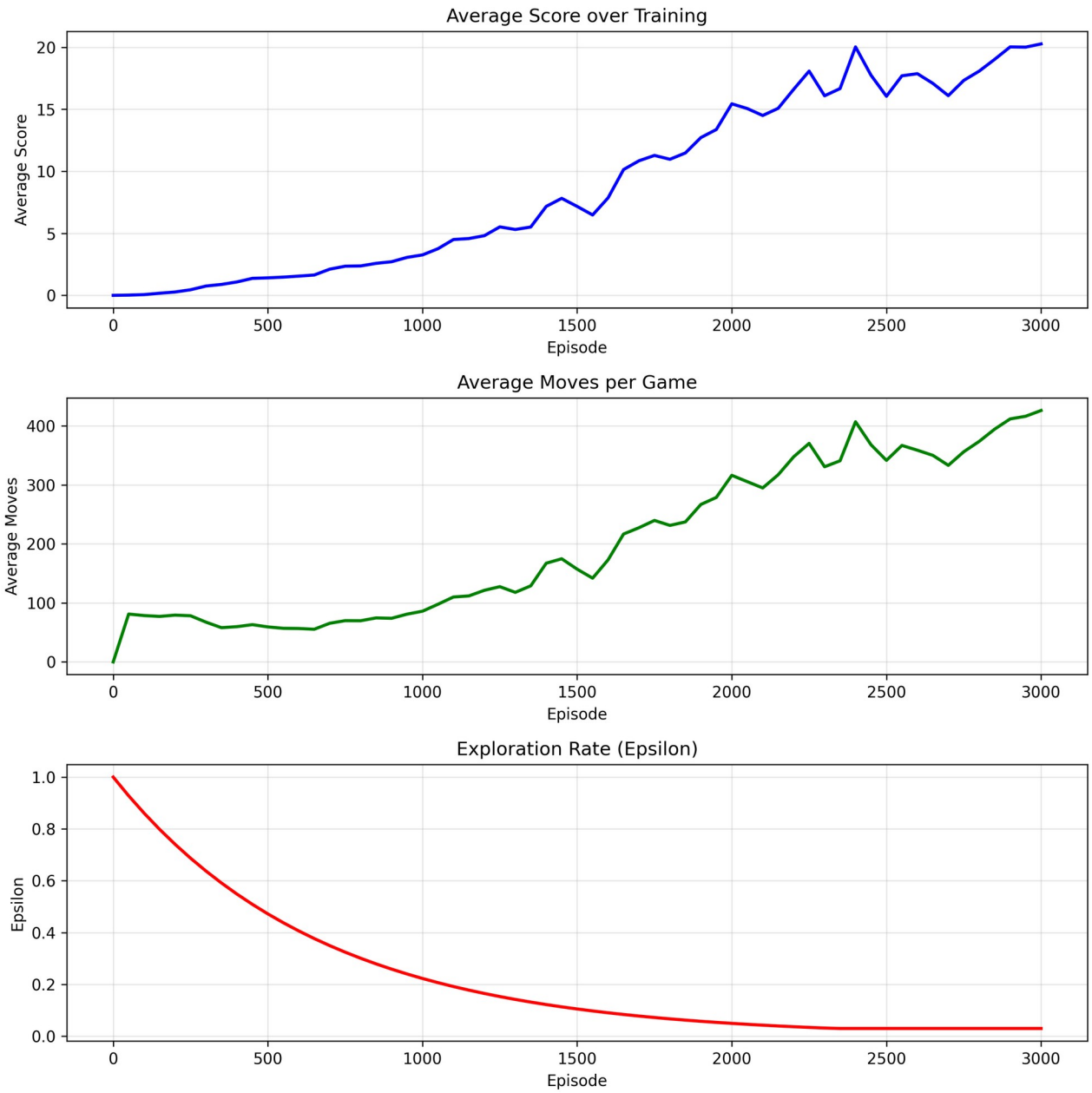
Агент 1 -v2 ова е дотрениранатата верзија на агент 1 која ја продолжува тренинг фазата за уште 1000 епизоди но со мала вредност на епсилон. Според графикот се гледа дека агентот со скоро 0 како вредност за епсилон. Ова покажува дека агентот подобро преформира но за поточна споредба ќе мора да ја видиме тест фазата на овие два агента.



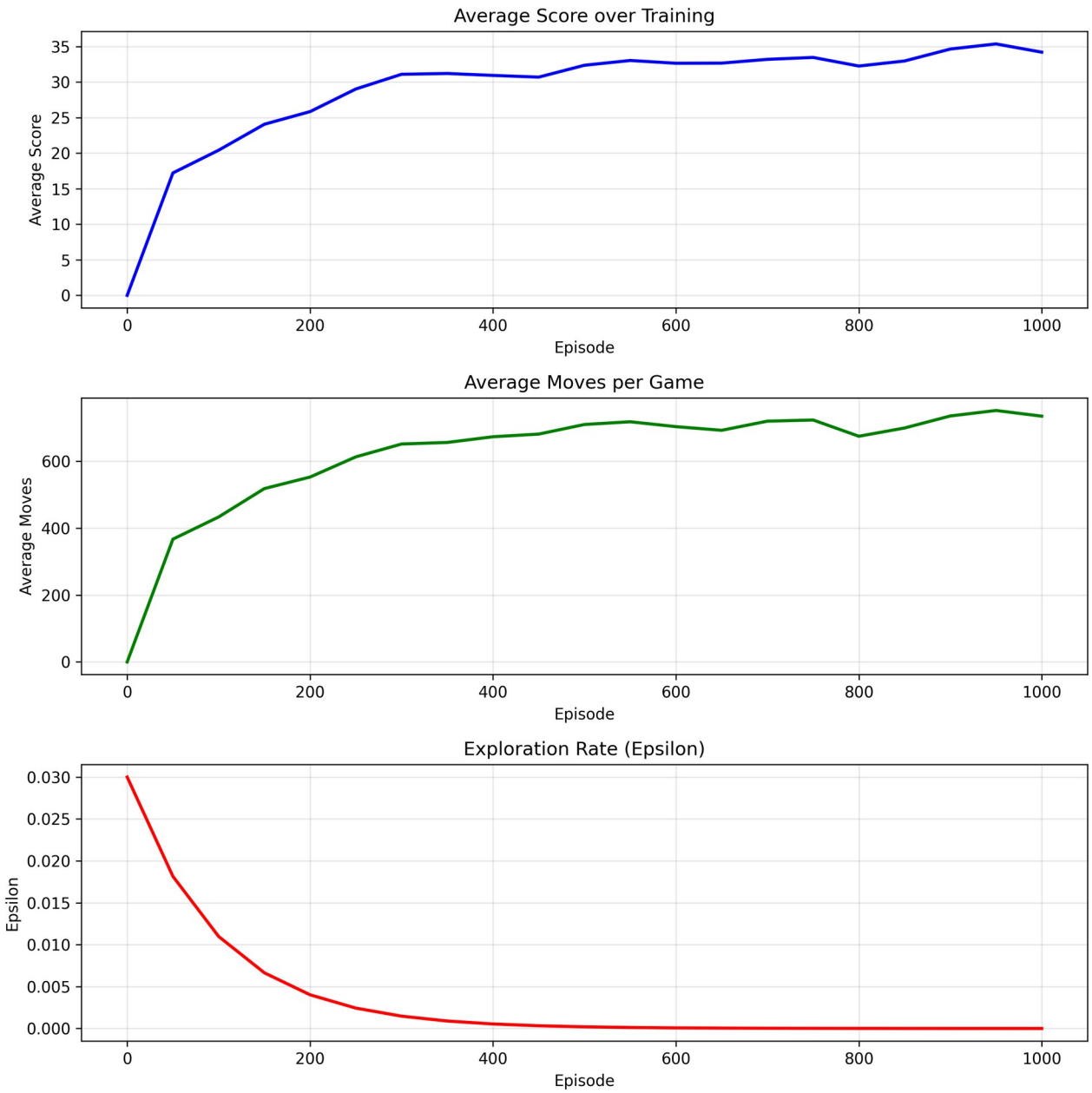
сл. бр. 3 Агент 2



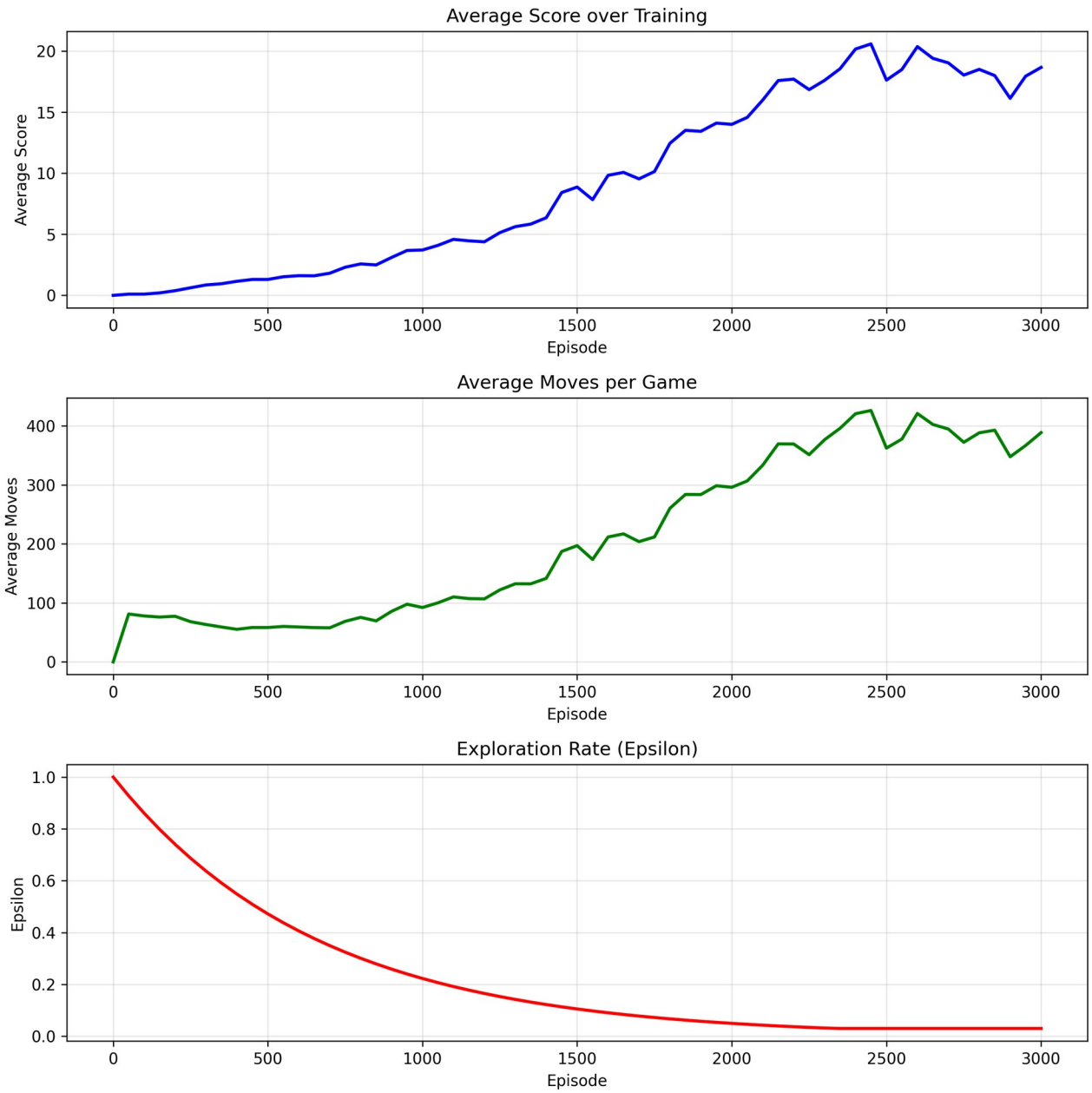
сл. бр.4 Агент 2-v2



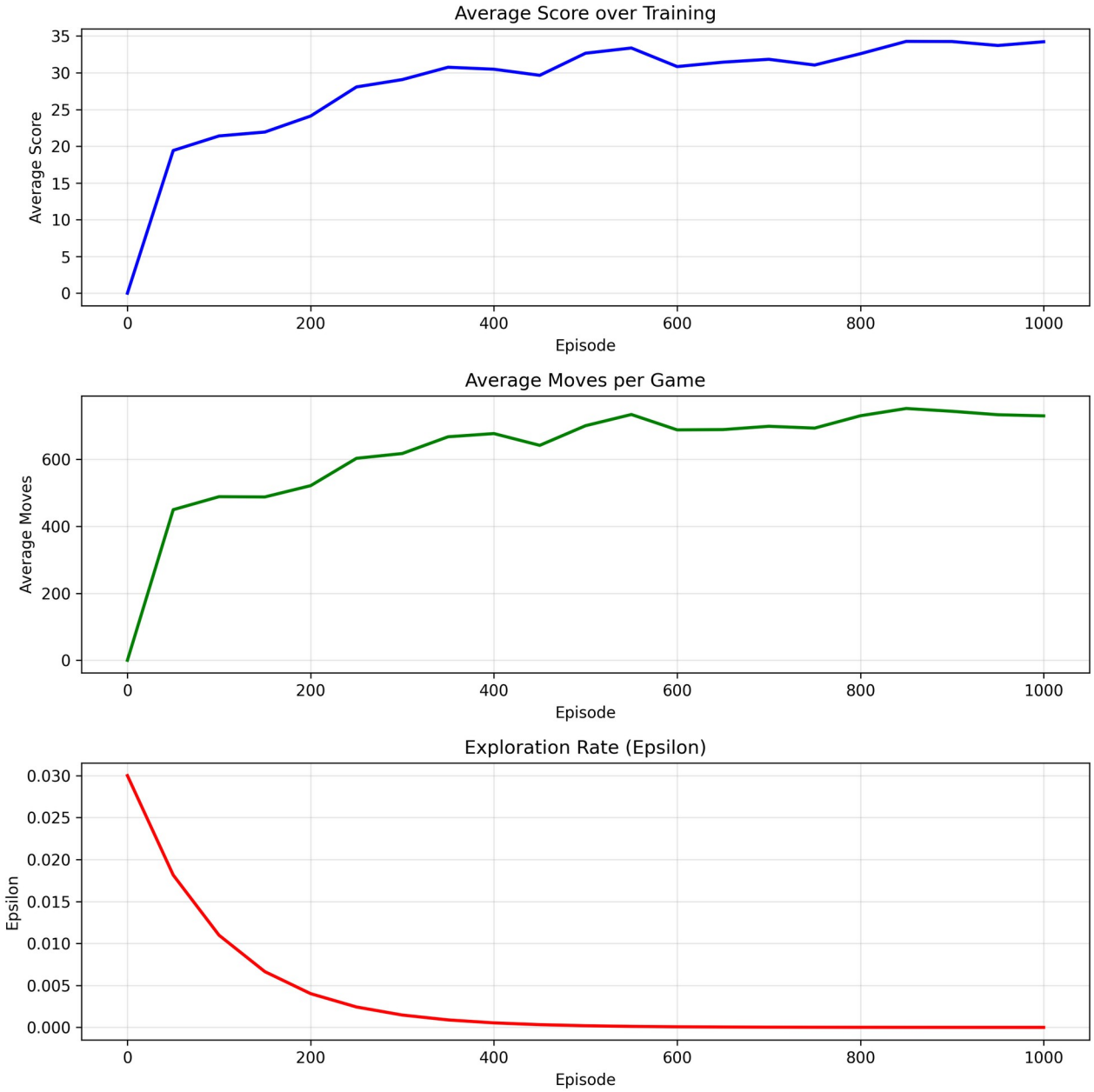
сл. бр.5 Агент 3



сл. бр. 6 Агент 3-v2



сл. бр. 7 Агент 4



сл. бр. 8 Агент 4-v2

5.3 Заклучок од тренинг фазата

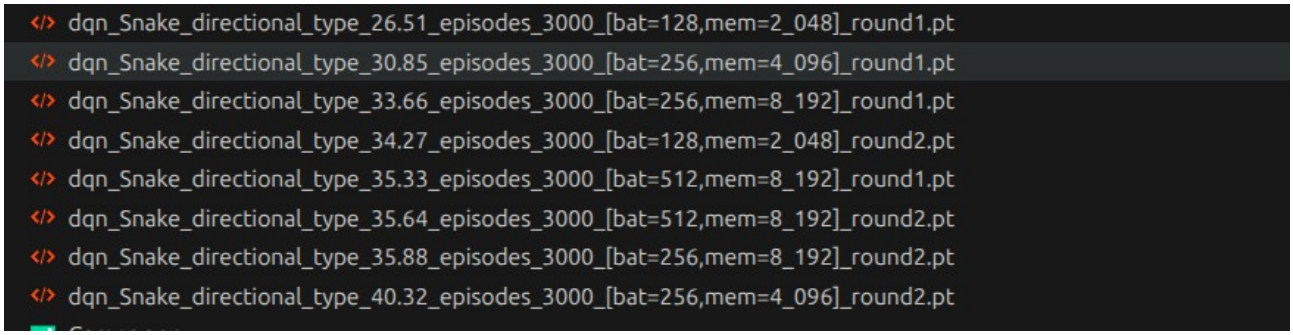
Кај сите агенти со тренинг од една рунда, се забележува нестабилно покачување на добиениот резултат, но тоа може да ја оправда вредноста на епсилон, иако мала. После 2000 епизоди, таа достигнува минимум 0,03 во првата рунда на тренирање. Иако ова изгледа незначајно, агентот во игра прави, на просек, помеѓу 300 и 400 чекори со таа вредност на епсилон, што значи дека има шанса барем 12 од тие 400 чекори да бидат рандомизирани. Со поголемата должина на змијата, исто така, се зголемува и шансата таа самата да се изеде.

Додека пак, кај агентите што се тренираат по втора рунда, се забележува постабилно покачување на резултатот. Тоа е благодарение на вредноста на епсилон, која станува скоро 0, и се намалуваат бројот на рандомизирани акции. Агентот скоро секогаш ја бира најдобрата акција.

6. Експериментални резултати од тест фазата

6.1.Резултати од тест фазата

Агентите за време на тест фазата играат 1000 игри и за тие игри добиваат просечен резултат кој се впишува во нивното име при зачувување. Следува слика од зачуваните агенти и нивните просечни резултати на 1000 игри.

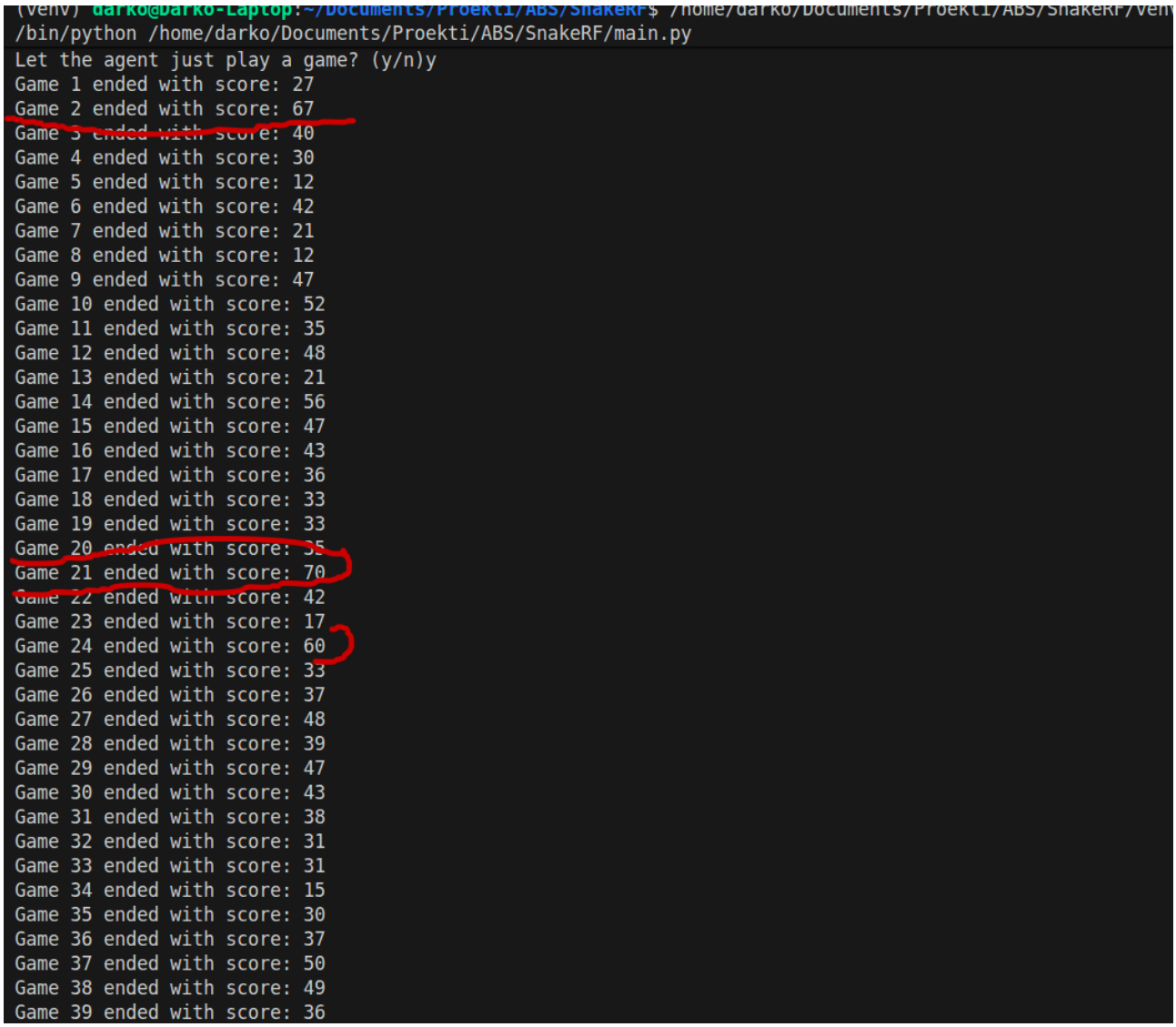


сл. бр.9

Коко што може да се забележи најуспешен агент е агентот 2-v2 со просечен резултат од 40 при 1000 игри изиграни. Од оваа слика може да забележиме исто така дека агентите бенифицирале од рунда два на тренирање кај сите има подобрување, освен кај агентот со најголема batch size 512 кој има минимално подобрување во рунда два. Ова ни кажува дека балансот помегју меморија и batch големина е побитен за добри преформанси на агентот.

6.2 Најдобриот агент во акција

Најдобриот агент кој има просечен резултат од 40 изигра 50 парти од играта Snake да ги погледнеме резултатите.



сл. бр.10

Најдобриот резултат кој агентот го постигнал е рекордни 70 јаболки изедени, но може да се забележи дека резултатот многу варира вомоменти има и катастрофално ниски 12 поени. Што значи ова? При обзервација на начинот на играње од страна на агентот е забележано често заробувачко движење. Што се мисли под заробувачо движење агентот сам се заробува во круг од своето тело, ова често се случува кога јаболкото се наогја на спротивната насока од движењето на агентот и неговото тело исто така му пречи. Зошто се случува ова? Агентот учи да оди во насока на јаболкото, ова е супер стратегија но само во раната фаза на играта кога змијата е кратка и не си попречува со своето тело, но во касните фази на играта ова е проблематична тактика.

7. Заклучок

Трудот успешно го имплементира и евалуира Deep Q-Learning агент за играта Snake со directional репрезентација од 18 карактеристики. Најдобриот агент (Агент 2-v2 со batch 256 и memory 4096) постигна просечен резултат од 40 поени на 1000 тест игри, со најдобар резултат од 70 јаболки. Двофазниот тренинг пристап (прва фаза со epsilon 1.0->0.03, втора фаза 0.03->0) покажа подобрување кај сите агенти. Клучен проблем е самозаробувачкото движење во касните фази, каде агентот се заробува во круг од своето тело при обид да ја достигне храната.

8. Изворен код

https://github.com/DDivanisov/SnakeRF